Non-monotone Spectral Projected Gradient Method for Semidefinite Program with Log-Determinant and $ell_{1}-$1Norm Terms (New Trends of Numerical Optimization in Advanced Information-Oriented Society)

AUTHOR(S):

Nakagaki, Takashi; Fukuda, Mituhiro; Yamashita, Makoto; Kim, Sunyoung

Non-monotone Spectral Projected Gradient Method for Semidefinite Program with Log-Determinant and ℓ_1-Norm Terms

Takashi Nakagaki,Mituhiro Fukuda, Makoto Yamashita
Department of Mathematical and Computing Science
Tokyo Institute of Technology

Sunyoung Kim
Department of Mathematics
Ewha W. University

Abstract

A variant of the spectral projected gradient (SPG) method proposed by Birgin, Martinez and Raydan is proposed to solve semidefinite programs with log-determinant and ℓ_1-norm terms. The SPG is modified in the orthogonal projection of the iterates onto the convex feasible set in order to obtain a cheap computation. Numerical results on the problems considered in the literature confirm that the implementation of the proposed method can be comparably faster than other well-known methods for similar problems.

1 Introduction

Let $\boldsymbol{x}$ be an n-dimensional random variable following a Gaussian distribution $\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$. We frequently are in situations where we want to estimate the mean $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$ of the distribution. If $\boldsymbol{x}^{(1)},\boldsymbol{x}^{(2)},\ldots,\boldsymbol{x}^{(m)}$ are i.i.d. samples from this distribution,

$$\hat{\boldsymbol{\mu}} := \frac{1}{m}\sum_{i=1}^{m}\boldsymbol{x}^{(i)} \qquad \text{and} \qquad \widehat{\boldsymbol{\Sigma}} := \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{x}^{(i)}-\hat{\boldsymbol{\mu}})(\boldsymbol{x}^{(i)}-\hat{\boldsymbol{\mu}})^T$$

are candidates for this mean and covariance matrix, respectively, if $m \gg n$. In the case $m < n$, we need to impose some "structure" or conditions over the covariance matrix to estimate it. The Gaussian graphical models [5] and sparse covariance selection [3] are one of classical solutions, where problems such as

$$\begin{cases} \max & -\mathrm{tr}(\widehat{\boldsymbol{\Sigma}}\boldsymbol{X}) + \log\det\boldsymbol{X} \\ \text{s.t.} & X_{ij} = 0,\ (i,j) \in \Omega \subseteq \{1,2,\ldots,n\}^2 \\ & \boldsymbol{X} \succ \boldsymbol{O}, \end{cases}$$

where $\boldsymbol{X} \succ \boldsymbol{O}$ means that $\boldsymbol{X}$ is a symmetric positive definite matrix, or

$$\begin{cases} \max & -\mathrm{tr}(\widehat{\boldsymbol{\Sigma}}\boldsymbol{X}) + \log\det\boldsymbol{X} - \rho\|\boldsymbol{X}\|_1 \\ \text{s.t.} & \boldsymbol{X} \succ \boldsymbol{O}, \end{cases}$$

are solved for a regularizing parameter $\rho > 0$. See d'Aspremont *et al.* [2] or Yuan and Lin [15] for early discussion on the relation between these two problems.

In this short note, we consider the extension of the above problems:

$$(\mathcal{P}) \qquad \begin{cases} \min & f(\boldsymbol{X}) := \mathrm{tr}(\boldsymbol{C}\boldsymbol{X}) - \mu\log\det\boldsymbol{X} + \mathrm{tr}(\boldsymbol{\rho}|\boldsymbol{X}|) \\ \text{s.t.} & \mathcal{A}(\boldsymbol{X}) = \boldsymbol{b}, \boldsymbol{X} \succ \boldsymbol{O}, \end{cases}$$

where $\boldsymbol{C}$, $\boldsymbol{X}$, $\boldsymbol{\rho} \in \mathbb{S}^n$, the space of $n \times n$ real symmetric matrices, $|\boldsymbol{X}| \in \mathbb{S}^n$ the matrix obtained by taking the absolute value of every element X_{ij} $(1 \le i,j \le n)$ of $\boldsymbol{X}$, and $\mathcal{A}$ a linear map

of $\mathbb{S}^n \to \mathbb{R}^m$. In $(\mathcal{P})$, $\boldsymbol{C}, \boldsymbol{\rho} \in \mathbb{S}^n, \mu > 0, \boldsymbol{b} \in \mathbb{R}^m$, and the linear map $\mathcal{A}$ given by $\mathcal{A}(\boldsymbol{X}) = (\mathrm{tr}(\boldsymbol{A}_1\boldsymbol{X}), \mathrm{tr}(\boldsymbol{A}_2\boldsymbol{X}), \ldots, \mathrm{tr}(\boldsymbol{A}_m\boldsymbol{X}))^T$, where $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m \in \mathbb{S}^n$, are given data.

The dual of $(\mathcal{P})$ is given by:

$$(\mathcal{D}) \qquad \begin{cases} \max & g(\boldsymbol{y}, \boldsymbol{W}) := \boldsymbol{b}^T\boldsymbol{y} + \mu \log\det(\boldsymbol{C} + \boldsymbol{W} - \mathcal{A}^T(\boldsymbol{y})) + n\mu - n\mu\log\mu \\ \text{s.t.} & (\boldsymbol{y}, \boldsymbol{W}) \in \mathcal{F}, \end{cases}$$

where $\mathcal{F} := \widehat{\mathcal{W}} \cap \widehat{\mathcal{F}}$, $\widehat{\mathcal{W}} := \mathbb{R}^m \times \mathcal{W}$, $\widehat{\mathcal{F}} := \{(\boldsymbol{y}, \boldsymbol{W}) \in \mathbb{R}^m \times \mathbb{S}^n \mid \boldsymbol{C} + \boldsymbol{W} - \mathcal{A}^T(\boldsymbol{y}) \succ \boldsymbol{O}\}$, $\mathcal{W} := \{[\boldsymbol{W}]_{\leq\boldsymbol{\rho}} : \boldsymbol{W} \in \mathbb{S}^n\}$, and finally, for matrices $\boldsymbol{W}, \boldsymbol{\rho} \in \mathbb{S}^n$, $[\boldsymbol{W}]_{\leq\boldsymbol{\rho}}$ is the matrix whose (i,j)th element is $\min\{\max\{W_{ij}, -\rho_{ij}\}, \rho_{ij}\}$.

We propose a variation of the projected gradient method originally presented by Birgin *et al.* [1] to the dual problem $(\mathcal{D})$. In order to apply our method, we assume three conditions on $(\mathcal{P})$ and $(\mathcal{D})$: (i) $\mathcal{A}$ is surjective, *i.e.*, the set of $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m$ is linearly independent; (ii) The problem $(\mathcal{P})$ has an interior feasible point, *i.e.*, there exists $\boldsymbol{X} \succ \boldsymbol{O}$ such that $\mathcal{A}(\boldsymbol{X}) = \boldsymbol{b}$; (iii) A feasible point for $(\mathcal{D})$ is given or can be easily computed. *i.e.*, there exists $\boldsymbol{y} \in \mathbb{R}^m$ and $\boldsymbol{W} \in \mathbb{S}^n$ such that $|\boldsymbol{W}| \leq \boldsymbol{\rho}$ and $\boldsymbol{C} + \boldsymbol{W} + \mathcal{A}^T(\boldsymbol{y}) \succ \boldsymbol{O}$. These assumptions are not strong as many applications satisfy these assumptions with slight modifications.

Many approximate solution methods for solving variants of $(\mathcal{P})$ have been proposed over the years. Lu [8] was one of earliers who considered a systematic approach to solve this problem as an mathematical optimization problem. The subsequent Adaptive Spectral Gradient (ASPG) method and the Adaptive Nesterov's Smooth (ANS) method [9] are one of the earlier methods which can handle large-scale problems. Ueno and Tsuchiya [11] proposed a Newton method by localized approximation of the relevant data. Wang *et al.* [13] considered a primal proximal point algorithm which solves semismooth subproblems by the Newton-CG iterates. Employing the inexact primal-dual path-following interior-point method, Li and Toh in [6] demonstrated that the computational efficiency could be increased, despite the known inefficiency of interior-point methods for solving large-sized problems. Yuan [16] also proposed an improved Alternating Direction Method (ADM) to solve the sparse covariance problem by introducing an ADM-oriented reformulation. For a more general structured models/problems, Yang *et al.* [14] enhanced the method in [13] to handle block structured sparsity, employing an inexact generalized Newton method to solve the dual semismooth subproblem. They demonstrated that regularization using $\|\cdot\|_2$ or $\|\cdot\|_\infty$ norms instead of $\|\cdot\|_1$ in $(\mathcal{P})$ are more suitable for the structured models/problems. Wang [12] first generated an initial point using the proximal augmented Lagrangian method, then applied the Newton-CG augmented Lagrangian method to problems with an additional convex quadratic term in $(\mathcal{P})$. Li and Xiao [7] employed the symmetric Gauss-Seidel-type ADMM in the same framework of [13]. A more recent work by Zhang *et al.* [17] shows that $(\mathcal{P})$ with simple constraints as $X_{ij} = 0$ for $(i,j) \in \Omega$ can be converted into a more computationally tractable problem for large values of $\boldsymbol{\rho}$. Among the methods mentioned here, only the methods discussed in [13, 14, 12] can handle problems as general as $(\mathcal{P})$.

In here, we consider a modification of the non-monotone spectral projected gradient method originally proposed by Birgin *et al.* [1] over the $(\mathcal{D})$ which guarantee a superior performance than applying directly to the primal problem $(\mathcal{P})$. A detailed description of the method and its convergence can be found in [10]. The main difference with the original Birgin *et al.*'s method lays in the projection of the next iterate. In there, the next iterate is orthogonally projected onto a closed convex set. In our method, we do not perform an exact projection. We define the next iterate, which should satisfy a non-monotone Armijo condition, by first projecting the next iterate over a box-type constraint $\widehat{\mathcal{W}}$ and then approximately project over an Linear Matrix Inequality (LMI) constraint $\widehat{\mathcal{F}}$. These projections are cheaper than performing an orthogonal projection over the intersection of these two convex sets $\widehat{\mathcal{W}} \cap \widehat{\mathcal{F}}$ which corresponds to our feasible convex set $\mathcal{F}$.

In the next section, we describe our algorithm, the results concerning the stopping criterion and its convergence. In Section 3, we show the performance of the implemented algorithm, DSPG compared to some well-known codes. Finally, Section 4 has some concluding remarks.

2 The Non-Monotone Spectral Projected Gradient Method for the Dual Problem and its Convergence

The variant of the non-monotone spectral projected gradient method over the dual problem $(\mathcal{D})$ is described in the following algorithm, where the notation $\boldsymbol{X}^k := \boldsymbol{X}(\boldsymbol{y}^k, \boldsymbol{W}^k) = \mu(\boldsymbol{C} + \boldsymbol{W}^k + \mathcal{A}^T(\boldsymbol{y}^k))^{-1}$ is used. Therefore, once the approximate solution of the dual $(\boldsymbol{y}^k, \boldsymbol{W}^k)$ is computed, the approximate solution of the primal problem $\boldsymbol{X}^k$ can be recovered.

Also, $\boldsymbol{P}_S$ denotes the projection onto a closed convex set S:

$$\boldsymbol{P}_S(\boldsymbol{x}) := \arg\min_{\boldsymbol{y}\in S} ||\boldsymbol{y} - \boldsymbol{x}||,$$

and

$$\begin{aligned} \nabla g(\boldsymbol{y}, \boldsymbol{W}) &:= (\nabla_{\boldsymbol{y}} g(\boldsymbol{y}, \boldsymbol{W}), \nabla_{\boldsymbol{W}} g(\boldsymbol{y}, \boldsymbol{W})) \\ &= (\boldsymbol{b} - \mu\mathcal{A}((\boldsymbol{C} + \boldsymbol{W} - \mathcal{A}^T(\boldsymbol{y}))^{-1}), \mu(\boldsymbol{C} + \boldsymbol{W} - \mathcal{A}^T(\boldsymbol{y}))^{-1}) \\ &= (\boldsymbol{b} - \mathcal{A}(\boldsymbol{X}(\boldsymbol{y}, \boldsymbol{W})), \boldsymbol{X}(\boldsymbol{y}, \boldsymbol{W})). \end{aligned}$$

Algorithm 2.1 *(Dual Spectral Projected Gradient Method)*

Step 0: Set parameters $\epsilon \geq 0,\ \gamma \in (0,1),\ \tau \in (0,1),\ 0 < \sigma_1 < \sigma_2 < 1,\ 0 < \alpha_{\min} < \alpha_{\max} < \infty$ and an integer parameter $M \geq 1$. Take the initial point $(\boldsymbol{y}^0, \boldsymbol{W}^0)$ which satisfy condition (iii), and an initial projection length $\alpha^0 \in [\alpha_{\min}, \alpha_{\max}]$. Set an iteration number $k := 0$.

Step 1: Compute a search direction (a projected gradient direction) for the stopping criterion

$$\begin{aligned} (\Delta\boldsymbol{y}^k_{(1)}, \Delta\boldsymbol{W}^k_{(1)}) &:= \boldsymbol{P}_{\widehat{\mathcal{W}}}((\boldsymbol{y}^k, \boldsymbol{W}^k) + \nabla g(\boldsymbol{y}^k, \boldsymbol{W}^k)) - (\boldsymbol{y}^k, \boldsymbol{W}^k) \\ &= (\boldsymbol{b} - \mathcal{A}(\boldsymbol{X}^k), [\boldsymbol{W}^k + \boldsymbol{X}^k]_{\leq\boldsymbol{\rho}} - \boldsymbol{W}^k). \end{aligned} \tag{1}$$

If $||(\Delta\boldsymbol{y}^k_{(1)}, \Delta\boldsymbol{W}^k_{(1)})||_\infty \leq \epsilon$, stop and output $(\boldsymbol{y}^k, \boldsymbol{W}^k)$ as the approximate solution.

Step 2: Compute a search direction (a projected gradient direction)

$$\begin{aligned} (\Delta\boldsymbol{y}^k, \Delta\boldsymbol{W}^k) &:= \boldsymbol{P}_{\widehat{\mathcal{W}}}((\boldsymbol{y}^k, \boldsymbol{W}^k) + \alpha^k\nabla g(\boldsymbol{y}^k, \boldsymbol{W}^k)) - (\boldsymbol{y}^k, \boldsymbol{W}^k) \\ &= (\alpha^k(\boldsymbol{b} - \mathcal{A}(\boldsymbol{X}^k)), [\boldsymbol{W}^k + \alpha^k\boldsymbol{X}^k]_{\leq\boldsymbol{\rho}} - \boldsymbol{W}^k). \end{aligned} \tag{2}$$

Step 3: Apply the Cholesky factorization to obtain a lower triangular matrix $\boldsymbol{L}$ such that $\boldsymbol{C} + \boldsymbol{W}^k - \mathcal{A}^T(\boldsymbol{y}^k) = \boldsymbol{L}\boldsymbol{L}^T$. Let θ be the minimum eigenvalue of $\boldsymbol{L}^{-1}(\Delta\boldsymbol{W}^k - \mathcal{A}^T(\Delta\boldsymbol{y}^k))\boldsymbol{L}^{-T}$. Then, compute

$$\overline{\lambda}^k := \begin{cases} 1 & (\theta \geq 0) \\ \min\left\{1, -\frac{1}{\theta} \times \tau\right\} & (\theta < 0) \end{cases}$$

and set $\lambda^k_1 := \overline{\lambda}^k$. Set an internal iteration number $j := 1$.

Step 3a: Set $(\boldsymbol{y}_+, \boldsymbol{W}_+) := (\boldsymbol{y}^k, \boldsymbol{W}^k) + \lambda^k_j(\Delta\boldsymbol{y}^k, \Delta\boldsymbol{W}^k)$.

Step 3b: If

$$\begin{aligned} g(\boldsymbol{y}_+, \boldsymbol{W}_+) &\geq \min_{0\leq h\leq \min\{k, M-1\}} g(\boldsymbol{y}^{k-h}, \boldsymbol{W}^{k-h}) + \gamma\lambda^k_j\nabla_{\boldsymbol{y}} g(\boldsymbol{y}^k, \boldsymbol{W}^k)^T\Delta\boldsymbol{y}^k \\ &+ \operatorname{tr}(\nabla_{\boldsymbol{W}} g(\boldsymbol{y}^k, \boldsymbol{W}^k)\Delta\boldsymbol{W}^k) \end{aligned}$$

is satisfied, then go to Step 4. Otherwise, choose $\lambda^k_{j+1} \in [\sigma_1\lambda^k_j, \sigma_2\lambda^k_j]$, and set $j := j + 1$, and return to Step 3a.

Step 4: Set $\lambda^k := \lambda_j^k$, $(\boldsymbol{y}^{k+1}, \boldsymbol{W}^{k+1}) := (\boldsymbol{y}^k, \boldsymbol{W}^k) + \lambda^k(\Delta \boldsymbol{y}^k, \Delta \boldsymbol{W}^k)$, $(\boldsymbol{s}_1, \boldsymbol{S}_1) := (\boldsymbol{y}^{k+1}, \boldsymbol{W}^{k+1}) - (\boldsymbol{y}^k, \boldsymbol{W}^k)$ and $(\boldsymbol{s}_2, \boldsymbol{S}_2) := \nabla g(\boldsymbol{y}^{k+1}, \boldsymbol{W}^{k+1}) - \nabla g(\boldsymbol{y}^k, \boldsymbol{W}^k)$. Let $b^k := \boldsymbol{s}_1^T \boldsymbol{s}_2 + \mathrm{tr}(\boldsymbol{S}_1 \boldsymbol{S}_2)$. If $b^k \geq 0$, set $\alpha^{k+1} := \alpha_{\max}$. Otherwise, let $a^k := \boldsymbol{s}_1^T \boldsymbol{s}_1 + \mathrm{tr}(\boldsymbol{S}_1 \boldsymbol{S}_1)$ and set $\alpha^{k+1} := \min\{\alpha_{\max}, \max\{\alpha_{\min}, -a^k/b^k\}\}$.

Step 5: Increase the iteration counter $k := k + 1$ and return to Step 1.

The fact that above algorithm converges to the correct optimal solution can be found in [10]. In there, the most relevant results are that the optimality condition can be tested by the fixed point condition for an arbitrary step length $\alpha > 0$ [10]:

Lemma 2.2 *$(\boldsymbol{y}^*, \boldsymbol{W}^*)$ is optimal for $(\mathcal{D})$ if and only if $(\boldsymbol{y}^*, \boldsymbol{W}^*) \in \mathcal{F}$ and*

$$\boldsymbol{P}_{\widehat{\mathcal{W}}}((\boldsymbol{y}^*, \boldsymbol{W}^*) + \alpha \nabla g(\boldsymbol{y}^*, W^*)) = (\boldsymbol{y}^*, \boldsymbol{W}^*) \tag{3}$$

for some $\alpha > 0$.

Also, as we can see from (2) that either the norm of the search direction $(\Delta \boldsymbol{y}^k, \Delta \boldsymbol{W}^k) = \boldsymbol{P}_{\widehat{\mathcal{W}}}((\boldsymbol{y}^k, \boldsymbol{W}^k) + \alpha^k \nabla g(\boldsymbol{y}^k, \boldsymbol{W}^k)) - (\boldsymbol{y}^k, \boldsymbol{W}^k)$ or the norm of the search direction for the step length $\alpha^k = 1$ (1) $(\Delta \boldsymbol{y}_{(1)}^k, \Delta \boldsymbol{W}_{(1)}^k) = \boldsymbol{P}_{\widehat{\mathcal{W}}}((\boldsymbol{y}^k, \boldsymbol{W}^k) + \nabla g(\boldsymbol{y}^k, \boldsymbol{W}^k)) - (\boldsymbol{y}^k, \boldsymbol{W}^k)$ can be used for the stopping criterion. This fact is guaranteed by the following lemma [10]:

Lemma 2.3 *The search direction $(\Delta \boldsymbol{y}^k, \Delta \boldsymbol{W}^k)$ is bounded by $(\Delta \boldsymbol{y}_{(1)}^k, \Delta \boldsymbol{W}_{(1)}^k)$. More precisely,*

$$\min\{1, \alpha_{\min}\} ||(\Delta \boldsymbol{y}_{(1)}^k, \Delta \boldsymbol{W}_{(1)}^k)|| \leq ||(\Delta \boldsymbol{y}^k, \Delta \boldsymbol{W}^k)|| \leq \max\{1, \alpha_{\max}\} ||(\Delta \boldsymbol{y}_{(1)}^k, \Delta \boldsymbol{W}_{(1)}^k)||. \tag{4}$$

Unfortunately, it is not possible to determine a complexity bound for the number of iterates of the above algorithm. However, we can, similarly to the original SPG [1], prove the asymptotic convergence of Algorithm 2.1 [10]:

Theorem 2.4 *Algorithm 2.1 with $\epsilon = 0$ stops in a finite number of iterations attaining the optimal value g^*, or generate a sequence $\{(\boldsymbol{y}^k, \boldsymbol{W}^k)\}$ such that*

$$\lim_{k \to \infty} |g(\boldsymbol{y}^k, \boldsymbol{W}^k) - g^*| = 0.$$

3 Numerical Experiments

The implementation of Algorithm 2.1 is called DSPG in this section. We conduct numerical experiments of DSPG over some test problems considered in the literature [6].

For this purpose, the DSPG is compared with the inexact primal-dual path-following interior-point method (IIPM) [6], the Adaptive Spectral Projected Gradient method (ASPG) [9], the Adaptive Nesterov's Smooth method (ANS) [9], and the QUadratic approximation for sparse Inverse Covariance estimation method (QUIC) [4].

We note that different stopping criteria are used in each of the aforementioned codes. They obviously affect the number of iterations and consequently the overall computational time. For a fair comparison, we set the threshold values for the IIPM, ASPG, ANS, and QUIC comparable to that of DSPG. More precisely, the stopping criteria of the DSPG was set to

$$||(\Delta \boldsymbol{y}_{(1)}^k, \Delta \boldsymbol{W}_{(1)}^k)||_\infty \leq \epsilon,$$

where $\epsilon = 10^{-5}$. For the IIPM, we employed

$$\max\left\{\frac{gap}{1 + |f(\boldsymbol{X}^k)| + |g(\boldsymbol{y}^k, \boldsymbol{W}^k)|}, pinf, dinf\right\} \leq \text{gaptol} := 10^{-6},$$

where *gap*, *pinf*, *dinf* were specified in [6], and for the ASPG and ANS, we used two thresholds $\epsilon_0 := 10^{-3}$ and $\epsilon_c := 10^{-5}$ such that $f(\boldsymbol{X}) \geq f(\boldsymbol{X}^*) - \epsilon_0$ and $\max_{(i,j)\in\Omega} |X_{ij}| \leq \epsilon_c$ [6]. The QUIC stops when $\|\partial f(\boldsymbol{X}^k)\|/\mathrm{Tr}(\boldsymbol{\rho}|\boldsymbol{X}^k|) < 10^{-6}$.

The DSPG was experimented with the following parameters: $\gamma = 10^{-4}$, $\tau = 0.5$, $0.1 = \sigma_1 < \sigma_2 = 0.9$, $\alpha_{\min} = 10^{-15} = 1/\alpha_{\max}$, $\alpha_0 = 1$, and $M = 50$. In the DSPG, the mexeig routine of the IIPM was used to reduce the computational time. All numerical experiments were performed on a computer with Intel Xeon X5365 (3.0 GHz) with 48 GB memory using MATLAB.

We set the initial solution as $(\boldsymbol{y}^0, \boldsymbol{W}^0) = (\boldsymbol{0}, \boldsymbol{O})$, which satisfies the assumption (iii) for the instances tested in Subsections 3.1 and 3.2.

In the tables in Subsections 3.2, the entry corresponding to the DSPG under the column "primal obj." indicates the minimized function value ($\mathcal{P}$) for $\boldsymbol{X}^k$, while "gap" means the maximized function value ($\mathcal{D}$) for $(\boldsymbol{y}, \boldsymbol{W})$ minus the primal one. Therefore, it should have a minus sign. The entries for the IIPM, ASPG, and ANS under "primal obj." column show the difference between the corresponding function values and the primal objective function values of the DSPG. Thus, if this value is positive, it means that the DSPG obtained a lower value for the minimization problem. The tables also show the minimum eigenvalues for the primal variable, the number of (outer) iterations, and the computational time.

In order to measure the effectiveness of recovering the inverse covariance matrix $\boldsymbol{\Sigma}^{-1}$, we adopt the strategy in [6]. The normalized entropy loss (loss_E) and the quadratic loss (loss_Q) are computed as

$$\mathrm{loss}_E := \frac{1}{n}(\mathrm{tr}(\boldsymbol{\Sigma}\boldsymbol{X}) \log\det(\boldsymbol{\Sigma}\boldsymbol{X}) - n), \qquad \mathrm{loss}_Q := \frac{1}{n}\|\boldsymbol{\Sigma}\boldsymbol{X} - \boldsymbol{I}\|,$$

respectively. Notice that the two values should ideally be zero if the regularity term $\mathrm{tr}(\boldsymbol{\rho}|\boldsymbol{X}|)$ is disregarded in ($\mathcal{P}$).

Some other numerical results on the DSPG can be found in the extended version of this report [10].

3.1 Real Data Problems

Five problems from the gene expression data [6] were tested for performance comparison. Since it was assumed that the conditional independence of their gene expressions is not known, linear constraints were not imposed and $\boldsymbol{\rho} = \rho\boldsymbol{I}$ in ($\mathcal{P}$), where $\boldsymbol{I}$ denotes the identity matrix.

Figures 1-3 show the computational time (left axis) for each problem when ρ is changed. As ρ grows larger, the final solution $\boldsymbol{X}^k$ (of the DSPG) becomes sparser, as shown in the right axis for the number of nonzero elements of $\boldsymbol{X}^k$.

We can observe from these five cases shown in Figures 1–3 that DSPG is comparable to the IIPM in terms of the computational time. They depend on the value of the regularizing parameter $\rho > 0$. As expected, QUIC is faster on sparse problems (with ρ larger) and slow on dense problems (with ρ close to zero). We can clearly see that its computational time is proportional to the number of nonzero elements of the approximate solution $\boldsymbol{X}^k$.

3.2 Synthetic data

The numerical results on eight problems where $\boldsymbol{A} \in \mathbb{S}^n$ has a special structure such as diagonal band, fully dense, or arrow-shaped [6] are shown in Tables 1. For each $\boldsymbol{A}$, a sample covariance matrix $\boldsymbol{C} \in \mathcal{S}^n$ is computed from $2n$ i.i.d. random vectors selected from the n-dimensional Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{A}^{-1})$. In these experiments, we did not compared with QUIC.

We can see from Table 1 that DSPG is the fastest code to obtain a similar precision, excepting the "ar1" problem, the most difficult case. For this instance, IIPM is the winner.

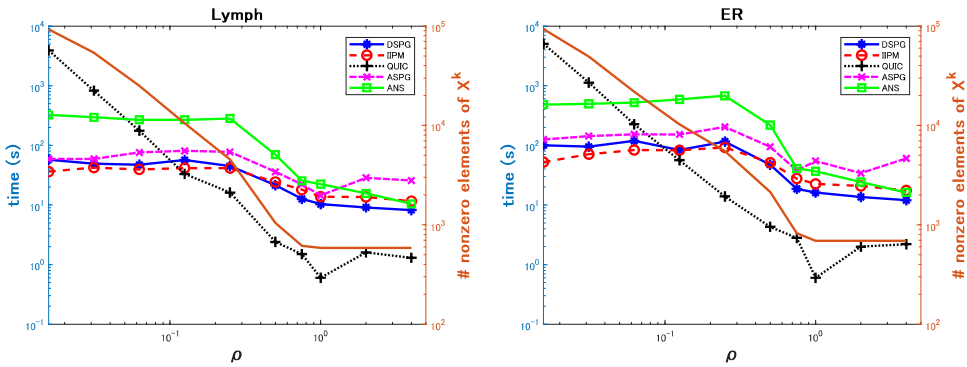

Figure 1: Computational time (the left axis) for the DSPG, IIPM, QUIC, ASPG, ANS on the problems "Lymph" ($n = 587$) and "ER" ($n = 692$) when $\rho \in [0.015625, 4]$; # of nonzero elements of $\boldsymbol{X}^k$ for the final iterate of DSPG (the right axis).

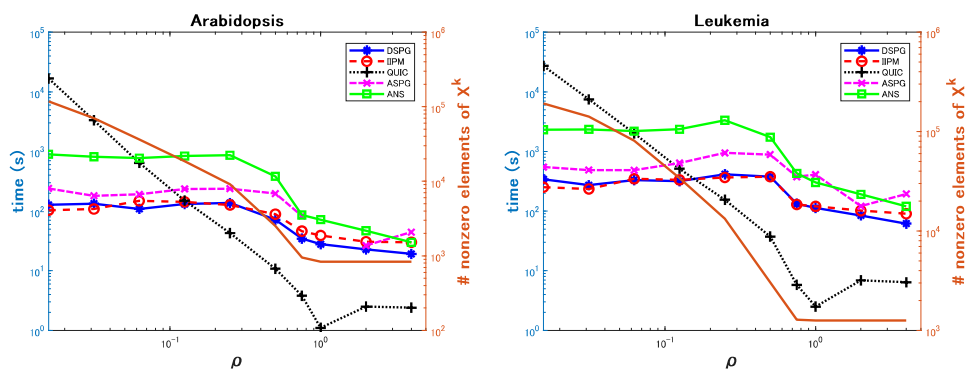

Figure 2: Computational time (the left axis) for the DSPG, IIPM, QUIC, ASPG, ANS on problems "Arabidopsis" ($n = 834$) and "Leukemia" ($n = 1255$) when $\rho \in [0.015625, 4]$; # of nonzero elements of $\boldsymbol{X}^k$ for the final iteration of DSPG (the right axis).

Table 1: Comparative numerical results for the DSPG, IIPM, ASPG and ANS on synthetic problems with n =2000.

problem	ρ	method	primal obj.	iter.	time (s)		
ar1	0.1	DSPG	3700.76285078	1353	3779.1	$\lambda_{\min}(\boldsymbol{X})$	$1.25e{-}06$
		(gap)	-0.02322140			loss_E	$3.3e{-}02$
		IIPM	-0.02303041	40	3195.5	loss_Q	$2.3e{-}01$
		ASPG	-0.02217276	3106	10681.1		
		ANS	-0.02217016	4448	13402.8		
ar2	0.1	DSPG	3026.63066861	45	128.2	$\lambda_{\min}(\boldsymbol{X})$	$2.58e{-}01$
		(gap)	-0.00890065			loss_E	$4.4e{-}02$
		IIPM	-0.00887081	13	362.1	loss_Q	$5.7e{-}03$
		ASPG	-0.00800225	55	225.4		
		ANS	-0.00791564	334	1149.0		
ar3	0.03	DSPG	2538.05453972	73	206.5	$\lambda_{\min}(\boldsymbol{X})$	$1.58e{-}01$
		(gap)	-0.00149801			loss_E	$2.5e{-}02$
		IIPM	-0.00138481	16	414.0	loss_Q	$5.3e{-}03$
		ASPG	-0.00070082	72	273.1		
		ANS	-0.00050606	376	1222.8		
ar4	0.01	DSPG	2257.32192663	39	111.2	$\lambda_{\min}(\boldsymbol{X})$	$2.10e{-}01$
		(gap)	-0.00015976			loss_E	$1.0e{-}01$
		IIPM	-0.00012374	15	382.5	loss_Q	$1.2e{-}02$
		ASPG	$+0.00048260$	35	159.7		
		ANS	$+0.00082845$	207	699.3		
Full	0.1	DSPG	2189.07471338	20	57.8	$\lambda_{\min}(\boldsymbol{X})$	$8.42e{-}01$
		(gap)	-0.33302912			loss_E	$7.9e{-}03$
		IIPM	-0.33297893	11	185.9	loss_Q	$2.1e{-}03$
		ASPG	-0.33297903	54	244.1		
		ANS	-0.33283013	40	150.5		
Decay	0.1	DSPG	2253.68280687	9	27.2	$\lambda_{\min}(\boldsymbol{X})$	$7.70e{-}01$
		(gap)	-0.01019772			loss_E	$1.5e{-}02$
		IIPM	-0.01011996	11	197.5	loss_Q	$3.6e{-}03$
		ASPG	-0.01003668	12	69.3		
		ANS	-0.00970895	32	126.3		
Star	0.1	DSPG	2204.50636824	32	91.2	$\lambda_{\min}(\boldsymbol{X})$	$2.50\text{-}2.51e{-}07$
		(gap)	-0.00115793			loss_E	$4.8e{-}03$
		IIPM	-0.00111934	12	201.1	loss_Q	$4.5e{-}01$
		ASPG	-0.00074587	31	160.2		
		ANS	-0.00044118	92	312.3		
Circle	0.05	DSPG	3506.82787956	510	1424.4	$\lambda_{\min}(\boldsymbol{X})$	$1.24e{-}06$
		(gap)	-0.00789762			loss_E	$3.5e{-}02$
		IIPM	-0.00781937	28	1483.4	loss_Q	$2.6e{-}01$
		ASPG	-0.00692232	751	2527.8		
		ANS	-0.00689055	1923	5865.4		

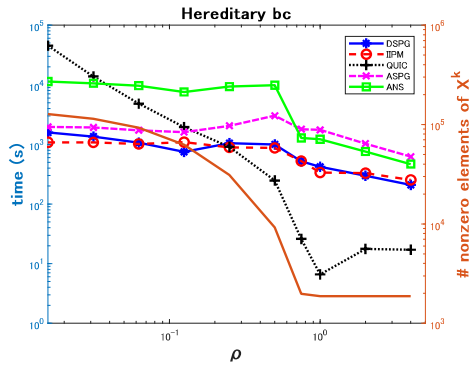

Figure 3: Computational time (the left axis) for the DSPG, IIPM, QUIC, ASPG, ANS on the problem "Hereditary bc" ($n = 1869$) when $\rho \in [0.015625, 4]$; # of nonzero elements of $\boldsymbol{X}^k$ for the final iteration of DSPG (the right axis).

4 Concluding Remarks

In this note, we have proposed a variant of the spectral gradient method applied to the dual problem of the semidefinite program with log-determinant and ℓ_1-norm terms. The method is efficient in terms of computational time since it avoids to perform an orthogonal projection onto the convex feasible set. Instead, it executes two projections onto convex sets which maintains feasibility. Convergence of the method can be proved [10]. The numerical experiments compared to similar codes show that it can be faster in some instances, specially compared to the state-of-art IIPM [6].

For further work, we can consider extending methods to more general cases such as minimization of quadratic convex functions [12] or some block diagonal structure considered on the matrix variable [13, 14].

Acknowledgement

M. F. research was partially supported by JSPS Grant-in-Aid for Scientific Research (C) No: 26330024, and by the Research Institute for Mathematical Sciences, a Joint Usage/Research Center located in Kyoto University. M. Y. research was partially supported by JSPS Grant-in-Aid for Scientific Research (C) No: 18K11176. And S. K. research was supported by NRF 2017-R1A2B2005119.

References

[1] E. G. Birgin, J. M. Martinez, and M. Raydan, "Nonmonotone spectral projected gradient methods on convex sets," *SIAM Journal on Optimization*, **10** (2000), pp. 1196–1211.

[2] A. d'Aspremont, O. Banerjee, and L. El Ghaoui, "First-order methods for sparse covariance selection," *SIAM Journal on Matrix Analysis and Applications*, **30** (2008), pp. 56–66.

[3] A. P. Dempster, "Covariance selection," *Biometrics*, **28** (1972), pp. 157–175.

[4] C.-J. Hsieh, M. A. Sustik, I. S. Dhillon, and P. Ravikumar, "QUIC: Quadratic approximation for sparse inverse covariance estimation," *Journal of Machine Learning Research*, **15** (2014), pp. 2911–2947.

[5] S. L. Lauritzen, *Graphical Models*, The Clarendon Press/Oxford University Press, Oxford, 1996.

[6] L. Li and K.-C. Toh, "An inexact interior point method for L_1-regularized sparse covariance selection," *Mathematical Programming Computation*, **2** (2010), pp. 291–315.

[7] P. Li and Y. Xiao, "An efficient algorithm for sparse inverse covariance matrix estimation based on dual formulation," *Computational Statistics & Data Analysis*, **128** (2018), pp. 292–307.

[8] Z. Lu, "Smooth optimization approach for sparse covariance selection," *SIAM Journal on Optimization*, **19** (2008), pp. 1807–1827.

[9] Z. Lu, "Adaptive first-order methods for general sparse inverse covariance selection," *SIAM Journal on Matrix Analysis and Applications*, **31** (2010), pp. 2000–2016.

[10] T. Nakagaki, M. Fukuda, S. Kim, and M. Yamashita, "Dual spectral projected gradient method for a log-determinant semidefinite problem," *Research Report,* **B-490** Department of Mathematical and Computing Science, Tokyo Institute of Technology, 2018.

[11] G. Ueno and T. Tsuchiya, "Covariance regularization in inverse space," *Quarterly Journal of the Royal Meteorological Society*, **135** (2009), pp. 1133–1156.

[12] C. Wang, "On how to solve large-scale log-determinant optimization problems," *Computational Optimization and Applications*, **64** (2016), pp. 489–511.

[13] C. Wang, D. Sun, and K.-C. Toh, "Solving log-determinant optimization problems by a Newton-CG primal proximal point algorithm," *SIAM Journal on Optimization*, **20** (2010), pp. 2994–3013.

[14] J. Yang, D. Sun, and K.-C. Toh, "A proximal point algorithm for log-determinant optimization with group lasso regularization," *SIAM Journal on Optimization*, **23** (2013), pp. 857–893.

[15] M. Yuan and Y. Lin, "Model selection and estimation in the Gaussian graphical model," *Biometrika*, **94** (2007), pp. 19–35.

[16] X. Yuan, "Alternating direction method for sparse covariance models," *Journal of Scientific Computing*, **51** (2012), pp. 261–273.

[17] R. Y. Zhang, S. Fattahi, and S. Sojoudi, "Large-scale sparse inverse covariance estimation via thresholding and max-det matrix completion," *Proceedings of Machine Learning Research*, **80** (2018), pp. 5766–5775.